IES de Teis
Avda. de Galicia, 101
36216 – Vigo

Tfno: 886 12 04 64
e-mail: ies.teis@edu.xunta.es
http://www.iesteis.es

Práctica Mappers y Reducers Python en YARN.

Objetivos

- Conocer y trabajar con mappers.
- Conocer y trabajar con reducers.

Fecha de entrega: viernes 12 de junio de 2024 hasta las 23:00 (hora del servidor)

Elabora y envía un documento PDF con tu nombre completo y DNI en la portada. Usa una nueva hoja por cada apartado, copia el enunciado y contesta.

En esta práctica realizaremos la implementación en Python de dos programas. El primero de ellos será un mapper que tiene como misión procesar parte de los datos en bruto. El segundo será un reducer que tiene como misión unir todos los datos previamente generados por los reducers.

El objetivo de esta práctica es familiarizarse con la filosofía MapReduce que es la usada en el campo de procesamiento distribuido en BigData. La idea es hacer la práctica siguiendo el enunciado y después plantearse donde habría que tocar, en el mapper o en el reducer, para realizar alguna modificación. La novedad de esta práctica es que la ejecutaremos en el clúster YARN que tengamos virtualizado.

Si aún no tienes un clúster HDFS+YARN funcionando adapta la práctica para poder ejecutarla en local como en la práctica anterior.

En esta práctica queremos obtener un listado de las distintas MACs que se han capturado así como el número de ocurrencias de cada una de ellas. Para ello os paso las capturas del día 24 de enero de 2024, el día en el que so convoqué a la primera sesión presencial. Los dispositivos capturados son aquellos que emiten beacons bluetooth regularmente como manos libres, cascos inalámbricos o relojes inteligentes. Si tu móvil no estaba activamente conectado en ese momento no aparecerá entre los resultados.

IES de Teis
Avda. de Galicia, 101
36216 – Vigo

Tfno: 886 12 04 64
e-mail: ies.teis@edu.xunta.es
http://www.iesteis.es

Enunciado

1.- Crea un archivo mapper.py que procese el archivo de la captura bluetooth que tienes en el aula virtual. Debe funcionar de una manera muy parecida al contador de palabras con la modificación que no todas las líneas son significativas. La única que nos interesa en este problema es la línea en la que aparece la MAC capturada en la que procesaremos única y exclusivamente la MAC, excluyendo la etiqueta "Dirección:". Como respuesta a esta pregunta muestra tu código comentado.

2.- (Opcional) Si tienes Hadoop funcionando no será necesario que pases por esta fase ya que se encargará Hadoop de hacerlo, puedes pasar al siguiente enunciado. Si tienes que ejecutar en local adapta tu código y aplica el archivo Python "ordenar.py" al archivo "salida_mapper.txt" para ordenar los resultados en un nuevo archivo "entrada_reducer.txt". Muestra la salida del comando "head -n 20 entrada_reducer.txt" para ver las 20 primeras líneas del resultado.

3.- Crea un archivo llamado reducer.py nos devuelva el número de apariciones de cada MAC. Como respuesta muestra tu código comentado.

4.- Prueba tu mapper y reducer en un clúster Hadoop con HDFS y YARN. Indica el comando que usas para ver los resultados y realiza una captura de pantalla de parte del resultado.

*Si no tienes el clúster Hadoop con HDFS y YARN funcionando modifica el enunciado de la práctica para usar archivos intermedios en su lugar. Igual que en la primera práctica.